UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER OF PATENTS AND TRADEMARKS
Washington, D.C. 20231
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/539,624	03/31/2000	Jerrie L. Coffman	219.38025X00	9576

20457 7590 04/24/2003

ANTONELLI TERRY STOUT AND KRAUS
SUITE 1800
1300 NORTH SEVENTEENTH STREET
ARLINGTON, VA 22209

EXAMINER	
ALI, SYED J	

ART UNIT	PAPER NUMBER
2127	2

DATE MAILED: 04/24/2003

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 09/539,624	Applicant(s) COFFMAN ET AL.	
	Examiner Syed J Ali	Art Unit 2127	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *31 March 2000* .

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-21* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-21* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

11)☐ The proposed drawing correction filed on _____ is: a)☐ approved b)☐ disapproved by the Examiner.

If approved, corrected drawings are required in reply to this Office action.

12)☐ The oath or declaration is objected to by the Examiner.

Priority under 35 U.S.C. §§ 119 and 120

13)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____ .

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

14)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).

a) ☐ The translation of the foreign language provisional application has been received.

15)☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☒ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) _____ .
4) ☐ Interview Summary (PTO-413) Paper No(s). _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other:

DETAILED ACTION

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-3, and 9-10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Goldberg et al. (USPN 5,319,782) (hereinafter Goldberg) in view of Valtin et al. (USPN 6.243.107) (hereinafter Valtin).

As per claim 1, Goldberg discloses a system comprising:

a shared resource (col. 3 lines 15-21, "Each thread constitutes a demand on subsystem resources", wherein the synchronization described therein is in reference to ensuring that the concurrently running threads do not conflict with each other);

multiple processors arranged to access said shared resource (this is inherent in Goldberg in that the reference relates to a typical data network, and it is well known that many servers as well as possible subsystems therein are capable of supporting multiple processors); and

an operating system configured to allow said multiple processors to perform work on said shared resource concurrently while supporting state changes or updates of said shared resources (col. 4 lines 45-60, "When a layer is invoked to perform a unit of work for a particular

connection...it will take the current state, service request input, and state transition table as parameters").

Goldberg does not specifically disclose said operating system comprising a synchronization algorithm for synchronizing multiple threads of operation with a single thread so as to achieve mutual exclusion between multiple threads performing work on said shared resource and a single thread updating or changing the state of said shared resource without requiring serialization of all threads.

Valtin discloses an operating system comprising a synchronization algorithm for synchronizing multiple threads of operation with a single thread so as to achieve mutual exclusion between multiple threads performing work on said shared resource and a single thread updating or changing the state of said shared resource without requiring serialization of all threads (col. 2 lines 5-38, "providing a summary of relevant changes to graphics state to each slave, thus guaranteeing correct state without requiring synchronization around state updates").

It is noted that the system of Valtin does not specifically relate to avoiding synchronization of threads in relation to a shared resource. However, as disclosed by the Goldberg there exists a reference that relates to shared resources in a multiprocessor system that support state changes. Therefore, it would have been obvious to one of ordinary skill in the art to utilize the method of Valtin of having a master thread perform all state changes and notify the other threads as to those state changes, since Goldberg has the deficiency of requiring synchronization among all threads. The method of Valtin avoids synchronizing threads, and therefore provides the benefit of ensuring that state changes and updates are done safely, since not more than one thread is making changes to the resource.

As per claim 2, Goldberg discloses the system as claimed in claim 1, wherein said shared resource includes work queues associated with a hardware adapter configured to send and receive message data to/from a remote system (col. 4 lines 3-8, "This permits the next waiting thread queued against the tie group to become the tie group owner and to be placed on the work queue of available threads").

As per claim 3, Valtin discloses the system as claimed in claim 2, wherein said synchronization algorithm is executed to synchronize any thread wishing to update or change a state of said shared resource with all the threads processing I/O operations on said shared resource (col. 2 lines 5-38, "providing a summary of relevant changes to graphics state to each slave, thus guaranteeing correct state without requiring synchronization around state updates", wherein the state changes made by the master thread are conveyed to all other threads in the system).

As per claim 9, Goldberg discloses the system as claimed in claim 2, wherein said synchronization algorithm is installed as part of a software driver module of an operating system (OS) kernel or an user-level application of said system (figs 4a-4d, wherein PASCAL code is shown that is capable of implementing the method described therein. Further, the program code shown in those figures could be modified to suit the needs of the system on which the algorithm is to be implemented, and therefore would have been an easy modification to include it as part of any software module, as claimed).

As per claim 10, "Official Notice" is taken that a shared resource includes ones of work queues, completion queues, FIFO queues, hardware adapters, I/O controllers and other memory elements of said system is well known and expected in the art. There are numerous synchronization algorithms that operate on shared resources, and these elements are well known to be among the types of resources that may be used by many concurrently running threads of execution.

3. Claims 4-5 are rejected under 35 U.S.C. 103(a) as being unpatentable over Goldberg in view of Valtin as applied to claim 1 above, and further in view of Vishkin (USPN 6,461,527).

As per claim 4, Valtin discloses the system as claimed in claim 1, wherein said synchronization algorithm is executed to allow an update thread to change the state or update said shared resource in exclusion of multiple worker threads (col. 2 lines 5-18, "the master thread to move between the processors to cause each slave thread to execute its graphics pipeline", "further providing a summary of relevant changes to graphics to each slave state", wherein the master thread performs all state updates and changes).

The modified Goldberg does not specifically disclose allowing worker threads to work concurrently while processing I/O operations in exclusion of an update thread when a state of said shared resource is not changing.

Vishkin discloses allowing worker threads to work concurrently while processing I/O operations in exclusion of an update thread when a state of said shared resource is not changing

(col. 3 lines 39-54, "all n threads are executed or run in parallel, thereby achieving an 'interthread' parallelism state").

It would have been obvious to one of ordinary skill in the art to combine the modified Goldberg with Vishkin since Goldberg does not explicitly define the capability of allowing concurrent execution of threads. Vishkin makes up for this deficiency by allowing multiple threads to execute concurrently on a shared resource, while still protecting the integrity of the resource by synchronizing all threads of execution.

As per claim 5, Vishkin discloses the system as claimed in claim 4, wherein said synchronization algorithm is executed to support a worker thread operation for processing simultaneous I/O operations on said shared resource while concurrently supporting an update thread operation for updating or changing the state of said shared resource (col. 3 lines 39-54, "all n threads are executed or run in parallel, thereby achieving an 'interthread' parallelism state", wherein Vishkin supports letting as many threads as the system can schedule execute in parallel). Further, since Valtin discloses allowing a single thread to perform all update operations. The combination therein of Goldberg, Valtin, and Vishkin teach all the limitations of the claim.

4. Claims 6-7, and 20-21 are rejected under 35 U.S.C. 103(a) as being unpatentable over Goldberg in view of Valtin in view of Vishkin and further in view of Haber et al. (USPN 4,435,766) (hereinafter Haber).

As per claim 6, the modified Goldberg does not specifically disclose a system as claimed in claim 5, wherein said worker thread operation is invoked by one of an event and a user's request, and is performed by:

determining whether a lock is available;

if the lock is not available, waiting until the lock becomes available;

if the lock is available, seizing the lock while incrementing a count by a discrete constant to indicate the number of worker threads that are active, and then releasing the lock after the count has been incremented;

after the lock has been released, allowing multiple worker threads to process work concurrently;

determining next whether there is work to be processed;

if there is work to be processed, processing the work until there is no work to be processed; and if there is no work to be processed, decrementing the count by a discrete constant to indicate when all the worker threads are done with completion processing.

Haber discloses determining whether a lock is available (col. 3 lines 29-64, "A LOCK routine is used to request exclusive access to the resource");

if the lock is not available, waiting until the lock becomes available (col. 3 lines 29-64, "If the resource is not available...the PCB of that process is chained onto a queue of PCBs which have issued LOCKs for the same resource and are waiting for the resource to become available");

if the lock is available, seizing the lock while incrementing a count by a discrete constant to indicate the number of worker threads that are active (col. 3 lines 29-64, "If a resource is

available...the lock count LCKCNT associated with the resource is incremented by one for each nested lock request");

after the lock has been released, allowing multiple worker threads to process work concurrently (Vishkin teaches this as discussed above in reference to claims 4 and 5; col. 3 lines 39-54, "all n threads are executed or run in parallel, thereby achieving an 'interthread' parallelism state");

determining next whether there is work to be processed (fig. 7 elements 202 and 208, wherein the USE function executes any work needing to be done by the program);

if there is work to be processed, processing the work until there is no work to be processed; andif there is no work to be processed, decrementing the count by a discrete constant to indicate when all the worker threads are done with completion processing (col. 3 lines 29-64, "The UNLOCK routine is used to decrement the lock count LCKCNT in the LCB for that resource").

As per claim 7, Haber discloses a system as claimed in claim 6, wherein said update thread operation is invoked by a user's request, and is performed by:

determining whether a lock is available (col. 3 lines 29-64, "A LOCK routine is used to request exclusive access to the resource");

if the lock is not available, waiting until the lock becomes available when released by any one of the worker threads (col. 3 lines 29-64, "If the resource is not available...the PCB of that process is chained onto a queue of PCBs which have issued LOCKs for the same resource and are waiting for the resource to become available");

if the lock is available, seizing the lock until the count becomes zero (0) to indicate that it is safe to update or change the state of said shared resource (col. 3 lines 29-64, "If the lock count LCKCNT goes to zero and no process is waiting for that resource then the resource is released"); and

after said shared resource has been updated, releasing the lock so as to allow either new worker threads to continue I/O operation processing or a different update thread to continue shared resource updating (col. 3 lines 29-64, "If the lock count LCKCNT goes to zero and no process is waiting for that resource then the resource is released").

Haber does not specifically disclose updating or changing the state of said resource.

Valtin discloses updating or changing the state of a resource (col. 2 lines 5-38, "providing a summary of relevant changes to graphics state to each slave, thus guaranteeing correct state without requiring synchronization around state updates").

As per claim 20, the modified Goldberg as discussed above (adding references Valtin, Vishkin, and Haber) discloses a process of synchronizing an update thread which updates a list of work queues with multiple worker threads which operate on items in the list of work queues in a multiprocessor system, comprising:

allowing a group of worker threads to concurrently access the list of work queues to process I/O operations in exclusion of an update thread, when states of the work queues are not changing (see claim 4);

incrementing a count of threads processing I/O operations each time a worker thread is running, while decrementing the count of threads processing I/O operations each time a worker thread is done processing I/O operations (see claim 6);

when the count of threads reaches a designated value indicating that no worker threads are running, allowing an update thread to access and update the list of work queues in exclusion of new worker threads from processing I/O operations (see claim 7); and

after the list of work queues is updated, allowing new worker threads to perform I/O operations until all worker threads are done processing I/O operations (see claim 7).

As per claim 21, it is rejected for similar reasons as stated above for claim 20.

5. Claims 8, 11-12, and 17-19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Goldberg in view of Valtin and further in view of Tillier (USPN 6,421,742).

As per claim 8, the modified Goldberg does not disclose a system as claimed in claim 2, further comprising data channels formed between said system and said remote system, via a switched fabric, and supported by the *"Virtual Interface (VI) Architecture Specification"* and the *"Next Generation Input/Output (NGIO) Specification"* for message data transfers between said system and said remote system.

Tillier discloses data channels formed between said system and said remote system, via a switched fabric, and supported by the *"Virtual Interface (VI) Architecture Specification"* and the *"Next Generation Input/Output (NGIO) Specification"* for message data transfers between said

system and said remote system (col. 5 lines 14-36, "The example embodiment and other embodiments of the invention can be implemented in conjunction with other types of switch fabric-based I/O architectures. The example embodiment NGIO uses a similar model for input/output data transfer as is specified by the VI architecture").

It would have been obvious to one of ordinary skill in the art to combine the modified Goldberg with Tillier since the modified Goldberg only speaks to synchronization of tasks in a multiprocessor system, but does not necessarily account for other types of networks, such as a switched fabric network. Tillier discloses a way of implementing such systems and thus would allow the modified Goldberg to be executed on a wider variety of systems.

As per claim 11, Tillier discloses a network, comprising:

a switched fabric (col. 5 lines 14-36, "The example embodiment and other embodiments of the invention can be implemented in conjunction with other types of switch fabric-based I/O architectures");

remote systems attached to said switched fabric (col. 2 lines 20-40, "The present invention is directed to emulation in an I/O unit when transferring data over a network" wherein it is inherent that a network would comprise remote systems); and

a host system comprising multiple processors; a host-fabric adapter provided to interface with said switched fabric and included work queues each configured to send and receive message data from a single remote system, via said switched fabric (fig. 1 element 101 – I_2O Host).

Tillier does not specifically disclose an operating system configured to allow said multiple processors to perform work on said work queues concurrently while supporting state

changes of said work queues, said operating system comprising a synchronization algorithm for synchronizing multiple threads of operation with a single thread so as to achieve mutual exclusion between multiple threads performing work on said work queues and a single thread changing the state of said work queues without requiring serialization of all threads.

The modified Goldberg as discussed regarding claim 1 discloses an operating system configured to allow said multiple processors to perform work on said work queues concurrently while supporting state changes of said work queues, said operating system comprising a synchronization algorithm for synchronizing multiple threads of operation with a single thread so as to achieve mutual exclusion between multiple threads performing work on said work queues and a single thread changing the state of said work queues without requiring serialization of all threads. The rejection of claim 1 therefore forms the basis for the rejection of this limitation.

As per claim 12, it is rejected for similar reasons as stated for claim 3 above. It is noted that claim 12 refers to "work queues" as opposed to a "shared resource", as in claim 3. However, as discussed regarding claim 10, a work queue is merely one example of a possible shared resource, and since the discussion of claim 3 covers the broader limitation of "shared resource", it covers the narrower limitation of "work queue" as well. This reasoning applies below as well to any claimed shared resource, which is narrower in scope than the "shared resource" discussed in claims 1-10.

As per claims 17-18, they are rejected for similar reasons as stated for claims 8-9 above, respectively.

As per claim 19, it is rejected for similar reasons as stated for claim 17 above.

6. Claims 13-14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Goldberg in view of Valtin in view of Tillier as applied to claim 11 above, and further in view of Vishkin.

As per claims 13-14, they are rejected for similar reasons as stated for claims 4-5 above, respectively. Further, the motivation for adding Vishkin to the Goldberg and Valtin is discussed in reference to claims 4-5, and the motivation for adding Tillier is discussed above in reference to claim 11.

7. Claims 15-16 are rejected under 35 U.S.C. 103(a) as being unpatentable over Goldberg in view of Valtin in view of Tillier in view of Vishkin as applied to claim 14 above, and further in view of Haber.

As per claims 15-16, they are rejected for similar reasons as stated for claims 6-7 above, respectively. Further, the motivation for adding Haber to the modified Goldberg is discussed in reference to claims 6-7.

Conclusion

8. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Syed J Ali whose telephone number is (703) 305-8106. The examiner can normally be reached on Mon-Fri 8-5:30, 1st Friday off.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John A Follansbee can be reached on (703) 305-8498. The fax phone numbers for the organization where this application or proceeding is assigned are (703) 746-7239 for regular communications and (703) 746-7238 for After Final communications.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is (703) 305-3900.

Syed Ali
April 18, 2003

JOHN FOLLANSBEE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100